(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 278 264 A2**

(12) EUROPEAN PATENT APPLICATION

(21) Application number: 88100674.6

(51) Int. Cl.⁴: G06F 13/36 , G06F 13/42

(22) Date of filing: 19.01.88

(30) Priority: 13.02.87 US 14757

(43) Date of publication of application:
17.08.88 Bulletin 88/33

(84) Designated Contracting States:
DE FR GB

(71) Applicant: International Business Machin s Corporation
Old Orchard Road
Armonk, N.Y. 10504(US)

(72) Inventor: Hoffman, Harrell
3509 Greenway
Austin Texas 78705(US)
Inventor: Wright, Charles
1204 Woodrock
Round Rock Texas 78681(US)

(74) Representative: Appleton, John Edward
IBM United Kingdom Limited Intellectual Property Department Hursley Park
Winchester Hampshire SO21 2JN(GB)

(54) Data processing system with overlap bus cycle operations.

(57) A data processing system including several devices connected to an asynchronous communications bus for communications between these devices. The communications bus includes a protocol that requires only a single device to regulate communication between devices at any one time. This regulating device is termed the bus master and the remaining devices are termed slaves. This protocol provides the capability for a slave device to indicate to the bus master that a new bus master is to be designated for a temporary communications. This communications with a different bus master then overlaps within the bus cycles of the designated bus master.

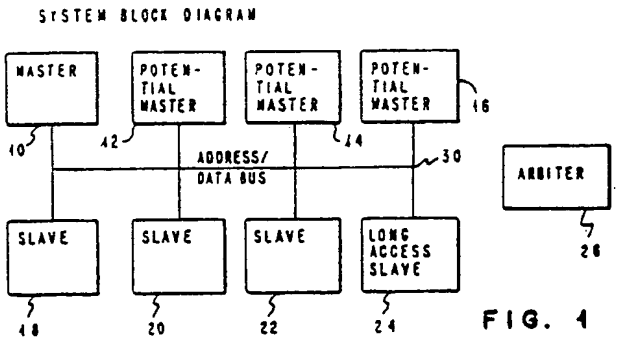


FIG. 1

EP 0 278 264 A2

DATA PROCESSING SYSTEM WITH OVERLAP BUS CYCLE OPERATIONS

This invention relates to data processing system communication and more specifically to a data processing system including a plurality of units connected to a common information bus for the transfer of information between the units.

Communications between data processing systems has become more complex with the increased capability of individual data processing units. This communication capability which is commonly termed networking. The specific area of interest for this invention is a single information bus connected to several processor units and several peripheral units. Traditionally, one processor unit is programmed to regulate the communications over the bus between the processing units and the peripheral units. In network terminology the communications controlling processor is termed the bus master and the remaining communication units termed slaves. In such a networking arrangement only the bus master may initiate any communications transactions on the bus. The slave units are only allowed to respond to communications from the bus master. A variation of this type of network provides for the bus master responsibility to be transferred among the processing unis. In this manner in any processor may address a peripheral device. If two processors desire to be bus master at the same time, a collision can occur on the information bus. Therefore, arbitration circuitry is provided to designate one of the communication bus requesting processors as the bus master.

An example of the above data processing system is disclosed in U.S. Patent Application 3,997,896, entitle "Data Processing System Providing Split Bus Cycle Operation", which discloses a data processing system including a common bus and several units connected to transfer information asynchronously wherein logic is provided for enabling a split bus cycle operation where the master unit requesting information from a slave during a first bus cycle may receive that information from the slave during a later bus cycle. This is accomplished by the slave at the later bus cycle requesting to a bus arbitrator that the slave be designated bus master in order that the slave may provide that information to the original bus master. Further disclosures of this type of bus architecture is contained in U.S. Patent 4,181,974, entitled "System Providing Multiple Outstanding Information Request" and U.S. Patent 4,236,203, entitled "System Providing Multiple Fetch Bus Cycle Operation".

In each of the above patents, the slave device is required to become bus master at the time the information that was originally requested becomes available. However, some slave devices do not include the capaability to function as a bus master. Therefore, it is an object of the present invention to provide a data processing system including a communications bus that enables a slave that cannot become bus master to provide requested information during later bus cycles.

In accordance with the present invention a data processing system including three devices connected to an asynchronous communications bus for the communications between the devices is disclosed. The communications bus includes a protocol that requires that only a single device regulate communications over the bus at any one time. The communications bus further includes circuitry for designating one of the devices to regulate the communications on this bus. The communications bus also includes circuitry to receive a request from another device to temporarily designate a second device to regulate communications and afterwards to redesignate the original regulating device.

In the preferred embodiment a communications interface is connected to several devices that are capable of regulating communications over a communications bus. Also connected to this communications bus are several peripheral devices. The interface to the communications bus for each of the devices connected includes circuitry to perform a protocol for the orderly transfer of information over the communications bus. In this communications circuitry is an arbitration circuit that designates one of the potential bus master devices as the bus master. The remaining devices are designated as slave devices that must respond to the bus master in accordance with the protocol. This communications bus circuitry further includes the capability to receive a request from any slave device to temporarily redesignate a potential bus master as a temporary bus master in order to establish a temporary communications exchange. After the temporary communications has been completed, the first bus master is then redesignated as bus master. In this embodiment, the bus master is required to provide a timing signal on the bus. When the designation of the bus master is changed, the first bus master will cease to provide this timing signal and the new bus master will begin providing this timing signal.

In the preferred embodiment the bus interface circuits at each of the connected devices includes circuitry to provide information from the bus to that device plus circuitry interconnecting to the other communications bus interfaces for providing information to the other devices. This interconnection of communication bus interfaces enables commu-

nication bus control signals to be interchanged among the bus interfaces to enable each device to communicate in accordance with the protocol. The present invention is directed specifically to the establishment of a temporary communication between two devices neither of which is the current bus master. This is termed an overlap bus communications cycle. In the present embodiment one of the slave devices includes the capability to perform complex operations that take a long time relative to the bus cycles and would normally require a long access time to complete. An original bus master may initiate a operation with the long access slave and then surrender control of the bus to a new bus master. When the long access slave is completed the operation and has the response for the original bus master, the long access slave provides a signal on the communications bus to alert the new bus master that a temporary communications cycle to the original bus master is required. In accordance with this invention the new bus master then surrenders control of the bus to the original bus master which then receives the response from the long access slave. After the response from the long access slave is received, the new bus master resumes control of the bus.

In this embodiment the protocol to facilitate this temporary communications requires the long access slave to provide a signal on the communications bus which is received by both the new bus master and the original bus master. The new bus master then issues a second timing signal which is received by the original bus master. The original bus master then issues a third timing signal which initiates the data transfer from the long access slave to the original bus master. After this temporary communications is completed, the original bus master ceases to provide this timing signal enabling the new bus master to reassert control over the bus.

In order that the invention may be fully understood a preferred embodiment will now be described with reference to the accompanying drawings in which:

Figure 1 is a block diagram of a communications bus connecting a bus master, several potential bus masters, and several slaves including a long access slave;

Figure 2 is a block diagram the bus interface circuitry for a bus master or potential bus master;

Figure 3 is a block diagram of the bus interface circuitry for a slave device;

Figure 4A is a timing diagram illustrating the initial communications timing between an original bus master and a long access slave;

Figure 4B is a timing diagram illustrating the redesignation of the original bus master as master and the completion of the temporary communications with the long access slave;

Figure 5 is a flow chart of the bus control sequencing performed by the original bus master;

Figure 6 is a flow chart illustrating the bus control sequence performed by the new bus master; and

Figure 7 is a flow chart illustrating the bus procedures performed by the long access slave device.

Figure 1 is a system block diagram illustrating the connection of a master device 10 to a communications bus 30. Also connected to the communication bus 30 are three potential master devices 12, 14 and 16 and four slave devices 18, 20, 22 and 24. Slave device 24 is a long access slave device. The normal communications transaction over bus 30 would require the bus master 10 providing a request for information from one of the slave devices 18, 20, 22, or 24 or from one of the potential bus masters 12, 14 or 16 and having that information provided in that same cycle. The bus master 10 may also initiate a long access operation with a long access slave 24 wherein the bus master 10 initially establishes the operation over the bus 30 with the long access slave 24 and then in a later bus cycle requests the information from the long access slave 24. However, this would require that the bus master 10 retain control of the bus throughout the communications with the long access slave 24.

An arbiter circuit 26 is also provided in Fig. 1. The arbiter circuit designates which of the potential bus master devices 12, 14 and 16 would become a new bus master after master device 10 has completed its communications task. Therefore, any of the potential bus master devices 12, 14 or 16 or the bus master 10 have the potential for communicating with any other device on the bus. The present invention solves the problem of how a bus master that initiates a long operation with a long access slave 24 and then surrenders control of the bus to a new bus master can receive its response. Without this invention, the original bus master cannot know when the operation by the long access slave 24 is complete until it again is designated bus master by the arbiter circuit 26. This disadvantage can seriously affect on system performance.

The present invention solves this problem by enabling the long access slave 24 to provide a signal on communications bus 30 indicating that the long operation has been completed and that a temporary communications cycle with the originating device is required. In response, the new bus master then temporarily surrenders control of the bus to the originating device or original bus master

so that the original bus master may receive the response from the long access slave 24.

Figure 2 illustrates in block diagram form the bus 30 interface of a bus master 10. This interface 10 will be connected to other processing capability or even a connection to another bus. This additional capability will originate and receive information transfers on the bus 30. While an interconnection is not shown, it should be apparent to those skilled in the art that such connections may be either directly to the address register 68, data register 70, and reply register 72 or through the control sequencer 50. This diagram is identical for the potential bus master 12, 14 and 16 of Fig. 1. The address and data are provided on line 52 to bus 30. Line 52 is connected to an address register 68, a data register 70 and a reply register 72 that is controlled by a bus control sequencer 50. Bus control sequencer 50 also interfaces to several control lines to bus 30. These include the DATA STROBE 54, the IO (Input/Output) cycle 56, the ACK (acknowledge) 58, the DATA GATE 60 and READY signal 62. Additionally, the bus controller sequencer 50 interfaces to arbiter 26 by providing a BUS REQUEST signal on line 64, a BUS GRANT signal on line 66, and a bus cycle COMPETE signal on line 47. The bus master 10 provides the IO cycle signal on line 56 and the DATA GATE signal on line 60 to control communications over the bus 30. The BUS REQUEST line 64 is used to request designation as bus master by arbiter 26. Designation as bus master is signified by the BUS GRANT line 66. The COMPETE line 67 is used to indicate to the arbiter 26 when the bus master is giving up bus control and is an indication to other potential bus masters as the period to request bus control from the arbiter 26.

A block diagram of the bus interface circuit for a slave device 24 is illustrated in Fig. 3. Line 52 containing address and data information connected to communication bus 30 is also connected to an address register 82, a data register 84 and a reply register 86. These registers 82, 84 and 86 are conneced to the bus control sequence 80 which is further connected to the bus control lines shown. The bus control lines include the DATA STROBE line 54, IO cycle line 56, the ACK line 58, the DATA GATE line 60 and the READY line 62.

Figure 4A illustrates the timing of the control line for a bus master initiating a long operation with a long access slave 24. Data is placed on line 102 at time 118 via the bus master. The bus master initiates the bus transfer by activating or lowering the IO cycle 104. A DATA STROBE is provided on line 106 by this original bus master. The slave device 24 acknowledges by providing a ACK signal on line 108. If the bus master is completed with its communications on bus 30, then the bus master drops COMPETE line 112 at time 124 indicating to the arbiter circuit 26 that a new bus master is to be designated. The READY line 114 and the DATA GATE line 116 are not activated during this transaction.

Figure 4B illustrates the timing when the new bus master has control of the bus. The new bus master controls the bus by lowering the IO cycle on line 134 when it is performing transactions over the bus. The communications bus protocol provides that when this new bus master raises the IO cycle line 134 (in other words there is no current bus transaction) a device such as the long access slave 24 requiring a temporary transfer may indicate to the new bus master that this long access slave 24 desires to provide a response to the original bus master. This protocol is the signal handshake sequence 154. The long access slave device 24 provides such an indication by activating the READY signal line 142. At this time the new bus master activates the ACK signal line 138 indicating that it has received the READY signal from the long access slave device and that it is willing to temporarily surrender control of the bus. It should be understood that the ACK signal is normally used by a device responding to a bus master during an active I/O cycle. The original bus master then indicates that it is taking control of the bus by activating the DATA GATE line 144. Therefore, at time 146 the original bus master has received control of the bus from the new bus master. The slave device 24 is placing data on line 132 at this time. The response data is then received by the original bus master on line 132 at time 146. The protocol to end this temporary communication is the signal handshake sequence 152. The long access slave 24 upon seeing that the DATA GATE signal 144 has been activated, deactivates the READY signal on line 142. After the original bus master has completed receiving the data on line 132, the original bus master indicates to the new bus master that control of the bus is being returned to the new bus master by deactivating the DATA GATE line 144. The new bus master then indicates that it is assuming control of bus by deactivating the ACK line 138. Therefore, at time 148, the new bus master is back in control of the bus. At time 150 the new bus master has initiated a new bus communications cycle.

Figure 5 is a flow chart of the bus control sequence for the original bus master executing this overlap communications operation. The original bus master starts at step 200 and determines if it needs to perform a cycle in 202. When it determines that a cycle is to be performed, the sequencer proceeds to step 204 to request control of the bus from the arbiter 26. Once control has been granted the sequencer proceeds to step 206 to

start an IO cycle on bus 30. As previously discussed this is initated by activating the IO cycle line 56 (Fig. 2). The sequencer then determines if an ACK is active in step 205. If an ACK has not been received, in other words the slave has not acknowledged the bus transmission, the sequencer goes to step 209 to indicate an error condition. If the ACK is received in step 205 the sequencer proceeds to step 207 to wait for a predetermined time. After this time the sequencer proceeds to step 208 to determine if the ACK signal remains. If an acknowledge (ACK) signal remains, the sequencer proceeds to step 224 to determine if a READY has been received on line 62. If not, the sequencer loops back to step 208. If so, the sequencer then proceeds to step 226 where the data gate line 60 is pulsed indicating that data is to be transmitted or received on the bus. The master then looks for the acknowledge (ACK) line to become inactive in step 228. Upon the inactivation of the acknowledge (ACK) line, the sequencer proceeds to step 222 to give up the bus by signaling the arbiter 26.

Returning to step 208, if the acknowledge (ACK) signal is not received, the sequencer proceeds to step 210 to finish the IO cycle. This would inidcate a long access requirement by the slave device being communicated with by the bus master. The sequencer then proceeds to step 212 to determine if any other devices on the bus are requesting access to the bus. If not, the sequencer proceeds to step 214 to look for the READY signal from the long access slave device 24 and continues in this loop if READY has not been received. If READY is received the sequencer proceeds to step 220. If another device is requesting access in step 212, the sequencer proceeds to step 216 to indicate to the arbiter 26 that it is giving up the bus. The sequencer then proceeds to step 218 where it monitors the ACK, READY and IO CYCLE lines waiting for the conditions illustrated in Fig. 4B to occur. When this occurs, the sequencer proceeds to step 220 to pulse the data gate line 60. Upon receiving the results from the long access slave 24, the sequencer of the original bus master then proceeds to step 222 to give up the bus by signaling the arbiter 26 and returning to the start position 200.

Figure 6 illustrates a flow chart of the bus control process for a new master. The new master starts at position 250 and proceeds to step 252 to determine if it in fact needs the bus. If it does not, it returns to the start position. If the new bus master requires the bus, then the new bus master sequencer proceeds to step 254 to request control of the bus from the arbiter 26. The sequencer then proceeds to step 256 to determine if the READY signal is present on the bus indicating that a long access slave device is requesting temporary communications with its original bus master. If no READY signal is present, then the new bus master sequencer proceeds to step 258 to determine if it needs to do a bus cycle. If a cycle is required, then the new bus master sequencer proceeds to step 260 to perform this cycle. If no cycle is required, the new bus master proceeds to stpe 262 to determine if any other devices connected to the bus are requesting control of the bus. If control of the bus by another device is being requested, the new bus master sequencer proceeds to step 264 to indicate to the arbiter 26 that it is giving up the bus. The sequencer then returns to the start position 250. If no other device is requesting control of the bus in step 262, the new bus master sequencer returns to step 256 to determine if a READY signal is present.

Returning to step 256, if a ready signal is present, the new bus master sequencer proceeds to step 268 to turn on the ACK line 58 to indicate to the original bus master that this new bus master is temporarily surrendering control of the bus. It should be understood that this ACK signal is not normally used by a bus master but used by a slave device in responding to a bus master.

The new bus master sequencer then proceeds to step 270 to determine if the DATA GATE signal has been activated and if the READY signal remains active. As long as either the READY or the DATA GATE signals are active, the new bus master sequencer will loop back to step 270. When the READY line and/or the DATA GATE line have been deactivated, the new bus master sequencer proceeds to step 272 to turn off the ACK line 58 indicating that the new bus master is again asserting control over the bus. The new bus master sequencer then returns to step 256.

It should also be understood that the procedures illustrated in Figs. 5 and 6 are stored in all potential bus master devices to be executed in the appropriate circumstances.

Figure 7 is a flow chart for the long access slave device 24 in performing long or short access operations. The sequencer for the slave device starts in position 300 and proceeds to step 302 to determine if a request has been received. If a request has been received, the slave device activates the ACK line 58 to indicate it has received this request. The slave sequencer proceeds to step 306 to determine if the operation requested is a long access operation. If not, the slave sequencer proceeds to step 324 to determine if the response is ready. The slave sequencer continues to loop at step 324 until the response does become ready. At which time it proceeds to step 326 to activate the READY line 62. The slave sequencer proceeds to step 328 to determine if the DATA GATE signal on line 60 has been received. If not, it waits in a loop

at step 328. When the DATA GATE signal is received, the sequencer proceeds to step 230 to send the reply and to turn off the ACK signal on line 58 and the READY signal on line 62. The sequencer then loops back to the start position 300.

Returning to step 306, if the slave sequencer determines that a request is a long access request, the sequencer proceeds to step 308 to turn off the ACK signal on line 58 indicating to the requesting bus master that this request is a long access request. The sequencer then proceeds to step 310 to determine when the response is ready. When this response does become ready, the sequencer proceeds to step 312 to determine if the IO cycle line is active. If the IO cycle line is active, the sequencer loops back to step 312. Once the IO cycle becomes inactive, the sequencer proceeds to step 314 to activate the READY line 62. The sequencer then determines if the IO cycle has been reactivated in step 316. If so, the sequencer turns off the READY signal on line 62 in step 318 and returns to step 312 to wait for the IO cycle to go inactive again. If the IO signal remains inactive in step 316, the sequencer proceeds to step 320 to determine if the DATA GATE signal on line 60 has been activated. If not, the sequencer loops back to step 316. If so, the sequencer then, in step 322, sends the response reply on the data lines 52 and turns off the READY signal on line 62 and returns to start position 300.

Claims

1. A data processing system including at least three devices connected to an asynchronous communications bus for intercommunications between said devices, said communications bus including a protocol requiring only a single device to regulate communication between devices at any one time, characterised in that said bus comprises:

means for designating a first device to regulate communications on said bus; and

means for another device to request said first device to designate a different device to temporarily regulate communications and to return bus regulation to the first device after said temporary communication.

2. A data processing system according to Claim 1 wherein said other device includes means to request said first device to temporarily designate a second device to regulate communications between said second device and said other device.

3. A data processing system according to Claim 2 wherein said communication bus includes means in said designated devices to provide at least one communications timing signal on said bus.

4. A data processing system according to Claim 3 wherein said timing means in said designated first device cease providing said timing signal upon designation of said second device.

5. A data processing system including at least a first, second and third device connected to an asynchronous communications bus for intercommunications between said devices, said communications bus including an arbitration circuit to designate a device as a first bus master to regulate communication between devices for a communications period and designating remaining devices as slave devices, characterised in that said bus comprises:

means connected to the slave devices for providing a signal indicating that a new bus master is to be designated temporarily for providing communications with the signaling device; and

means for redesignating another of said devices to be a new bus master and remaining devices as slave devices in response to said signal without any communications with said arbitration circuit.

6. A data processing system according to Claim 5 wherein said bus includes a plurality of timing means connected to all potential bus master devices for providing at least one communications timing signal on said bus, said timing means including means for stopping the timing signal when its connected device is not bus master.

7. A data processing system according to Claim 6 wherein said signalling slave device provides a signal to the first bus master indicating the request to designate the new bus master.

8. A data processing system according to Claim 7 wherein said first bus master includes means to provide a second signal on said bus indicating to the new bus master that the first bus master is relinquishing bus control.

9. A data processing system according to Claim 8 wherein said new bus master provides a third signal on said bus indicating to the first bus master and the signalling slave device that the new bus master has control of the bus and the communications with the signalling slave is to begin.

10. A data processing system according to Claim 9 wherein said new bus master removes the third signal upon completion of communications with the signalling slave indicating to the first bus master that control of the bus is being returned to the first bus master.

SYSTEM BLOCK DIAGRAM

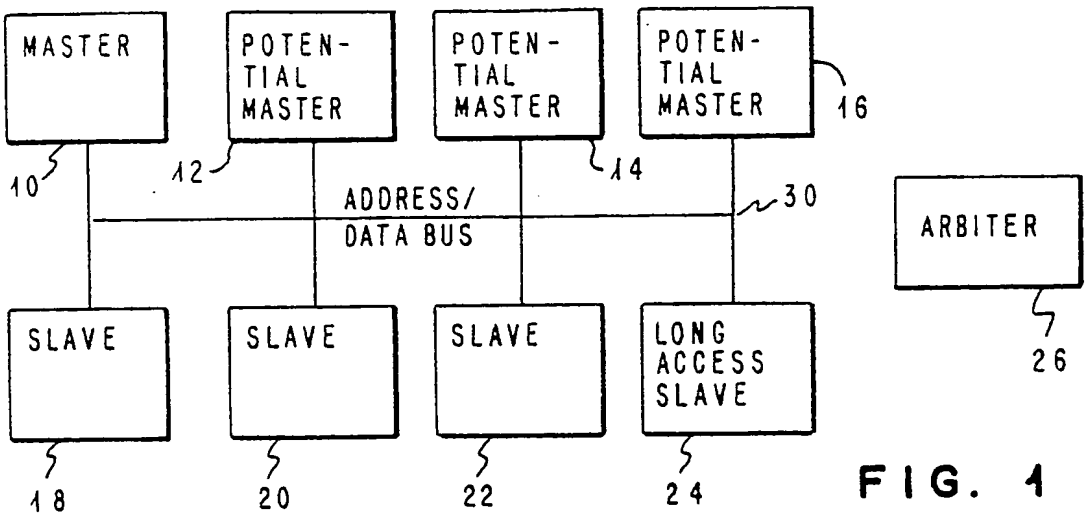


FIG. 1

BUS MASTER IMPLEMENTATION

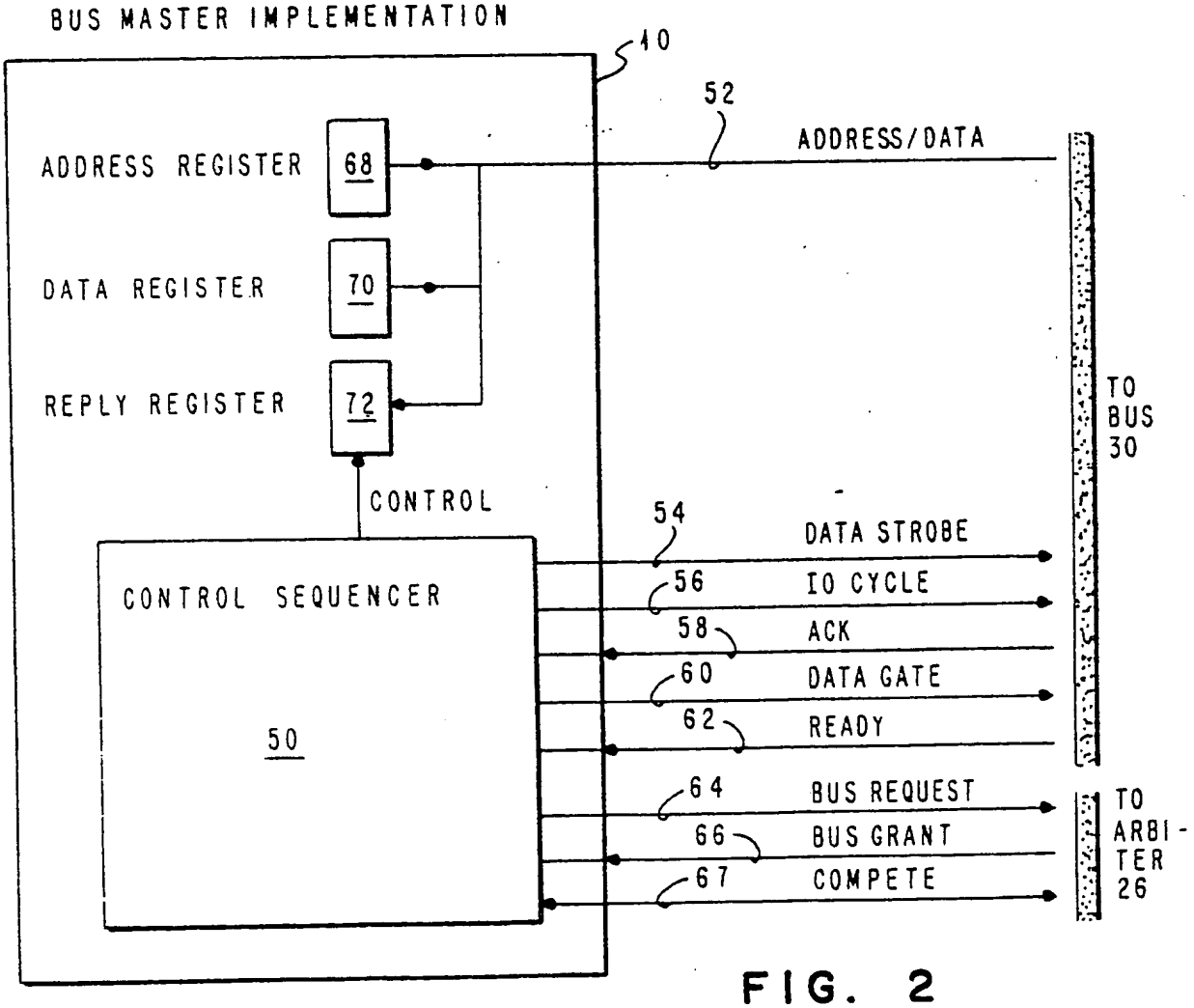


FIG. 2

BUS SLAVE IMPLEMENTATION

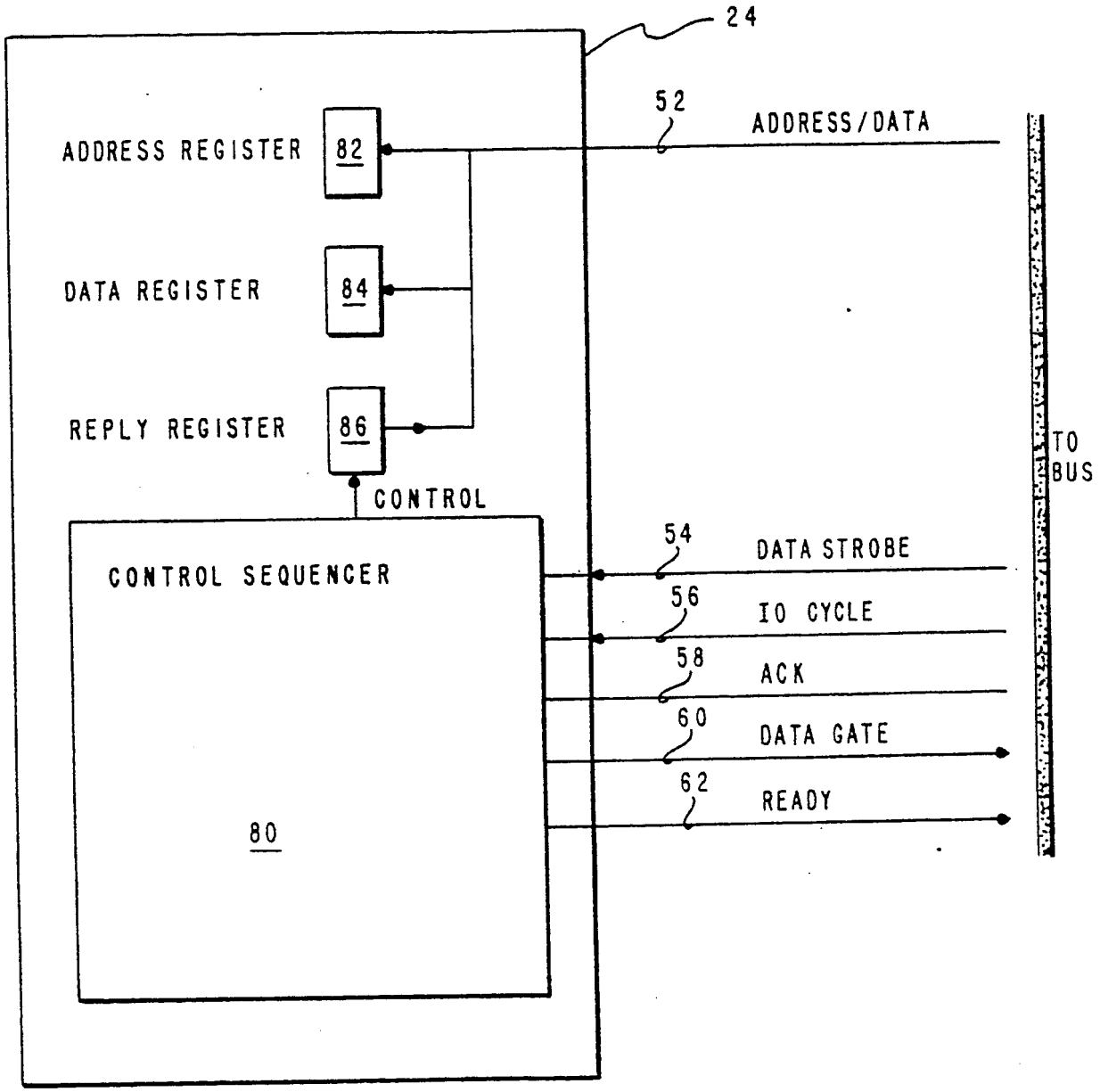


FIG. 3

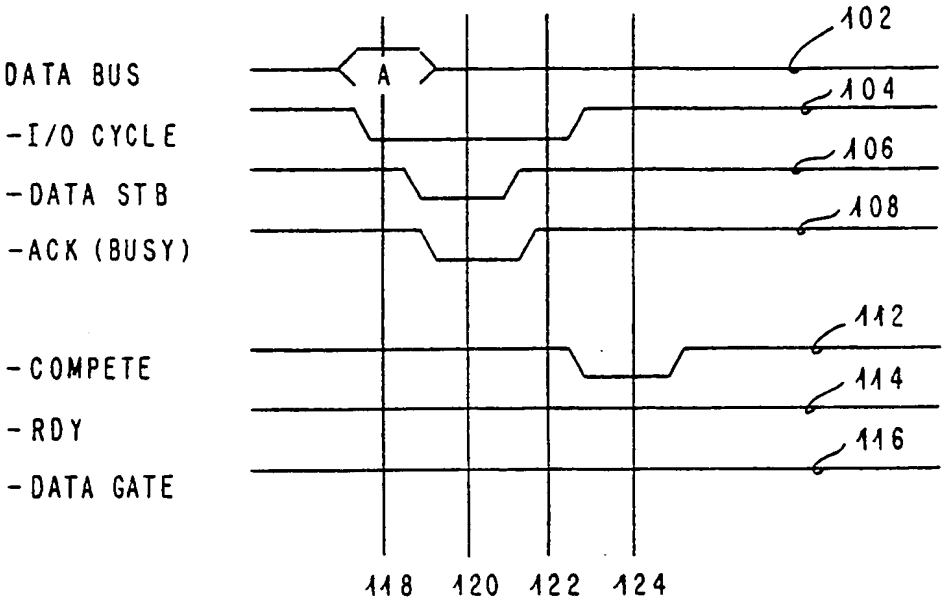

DATA BUS
-I/O CYCLE
-DATA STB
-ACK (BUSY)
-COMPETE
-RDY
-DATA GATE
A
102
104
106
108
112
114
116
118
120
122
124

FIG. 4A

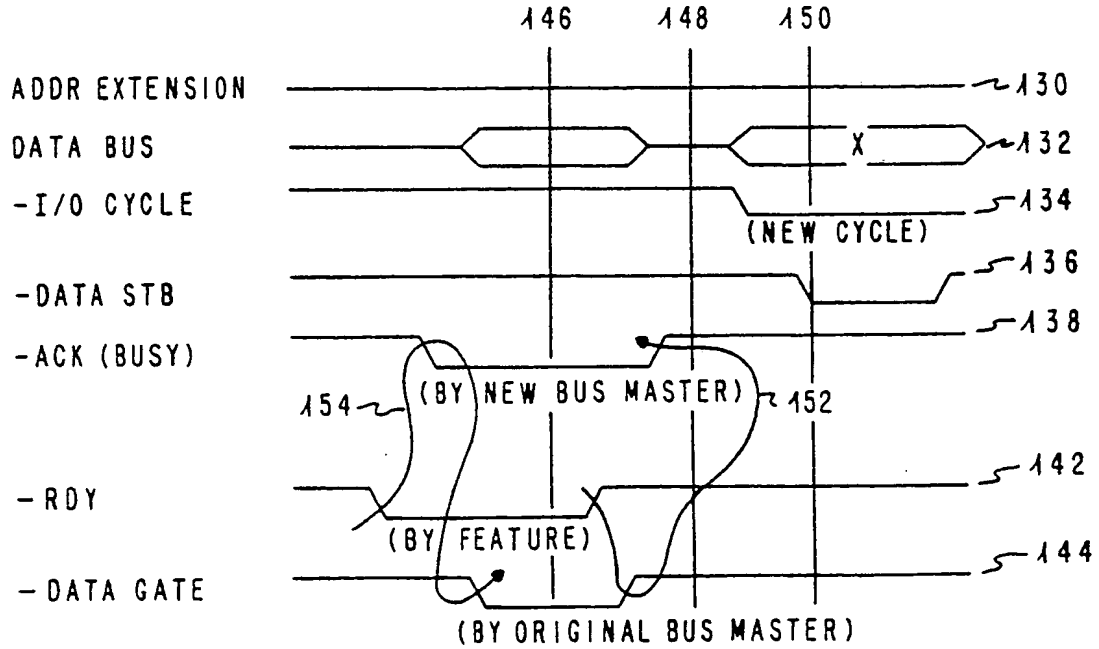
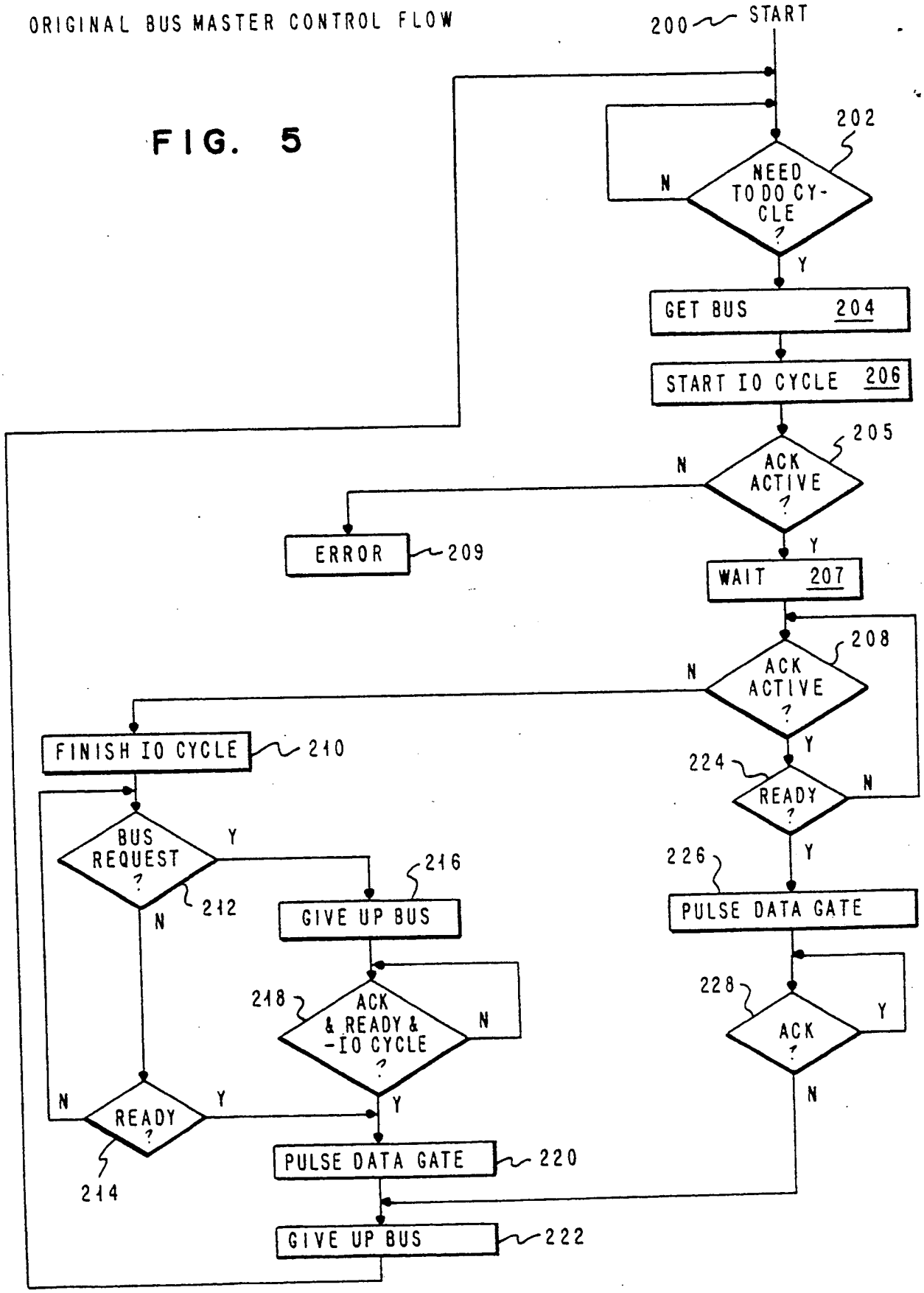

ADDR EXTENSION
DATA BUS
-I/O CYCLE
-DATA STB
-ACK (BUSY)
-RDY
-DATA GATE
146
148
150
130
132
X
134
(NEW CYCLE)
136
138
154
(BY NEW BUS MASTER)
152
142
(BY FEATURE)
144
(BY ORIGINAL BUS MASTER)

FIG. 4B

ORIGINAL BUS MASTER CONTROL FLOW

FIG. 5



NEW BUS MASTER CONTROL FLOW

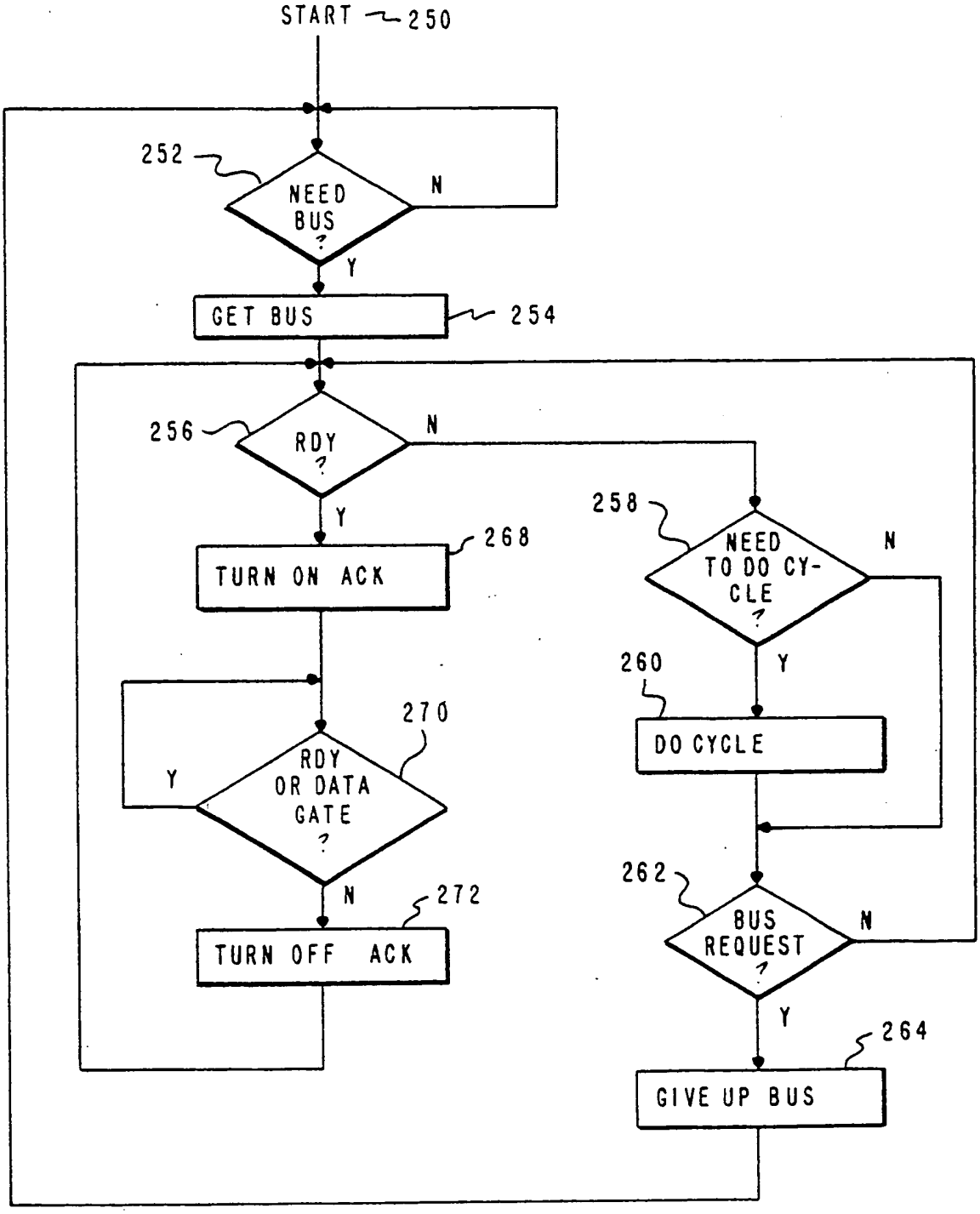


FIG. 6

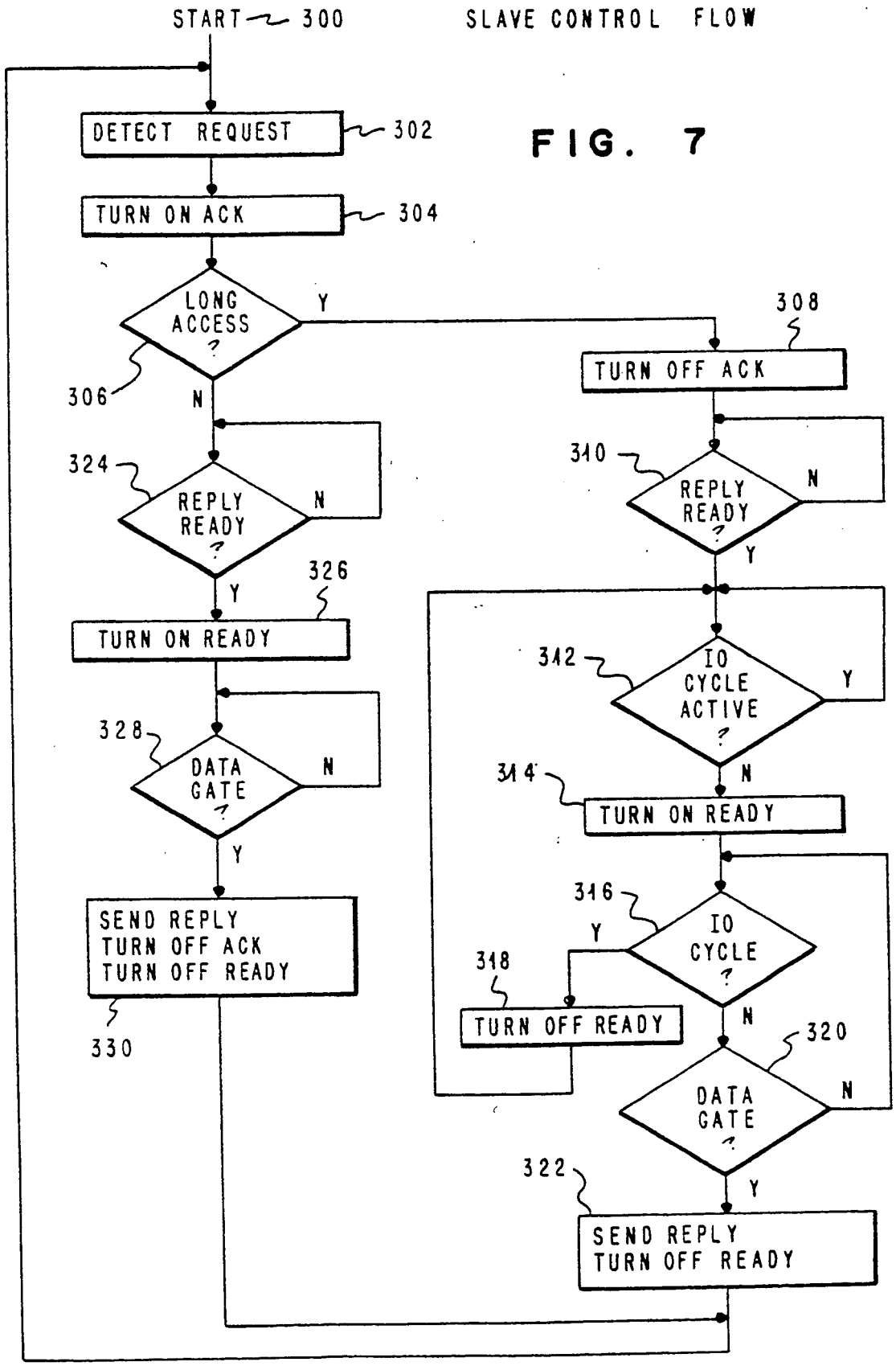

START
300
SLAVE CONTROL FLOW
DETECT REQUEST
302
TURN ON ACK
304
LONG ACCESS ?
Y
N
306
308
TURN OFF ACK
324
REPLY READY ?
N
Y
326
TURN ON READY
328
DATA GATE ?
N
Y
SEND REPLY
TURN OFF ACK
TURN OFF READY
330
310
REPLY READY ?
N
Y
312
IO CYCLE ACTIVE ?
Y
N
314
TURN ON READY
316
IO CYCLE ?
Y
318
TURN OFF READY
N
320
DATA GATE ?
N
Y
322
SEND REPLY
TURN OFF READY

FIG. 7

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 278 264 A3**

(12) EUROPEAN PATENT APPLICATION

(21) Application number: 88100674.6

(51) Int. Cl.5: G06F 13/36, G06F 13/42

(22) Date of filing: 19.01.88

(30) Priority: 13.02.87 US 14757

(43) Date of publication of application:
17.08.88 Bulletin 88/33

(84) Designated Contracting States:
DE FR GB

(88) Date of deferred publication of the search report:
09.01.91 Bulletin 91/02

(71) Applicant: International Business Machines Corporation
Old Orchard Road
Armonk, N.Y. 10504(US)

(72) Inventor: Hoffman, Harrell
3509 Greenway
Austin Texas 78705(US)
Inventor: Wright, Charles
1204 Woodrock
Round Rock Texas 78681(US)

(74) Representative: Johansson, Lars E.
IBM Svenska AB Intellectual Property
Department 4-01
S-163 92 Stockholm(SE)

(54) Data processing system with overlap bus cycle operations.

(57) A data processing system including several devices connected to an asynchronous communications bus for communications between these devices. The communications bus includes a protocol that requires only a single device to regulate communication between devices at any one time. This regulating device is termed the bus master and the remaining devices are termed slaves. This protocol provides the capability for a slave device to indicate to the bus master that a new bus master is to be designated for a temporary communications. This communications with a different bus master then overlaps within the bus cycles of the designated bus master.

SYSTEM BLOCK DIAGRAM

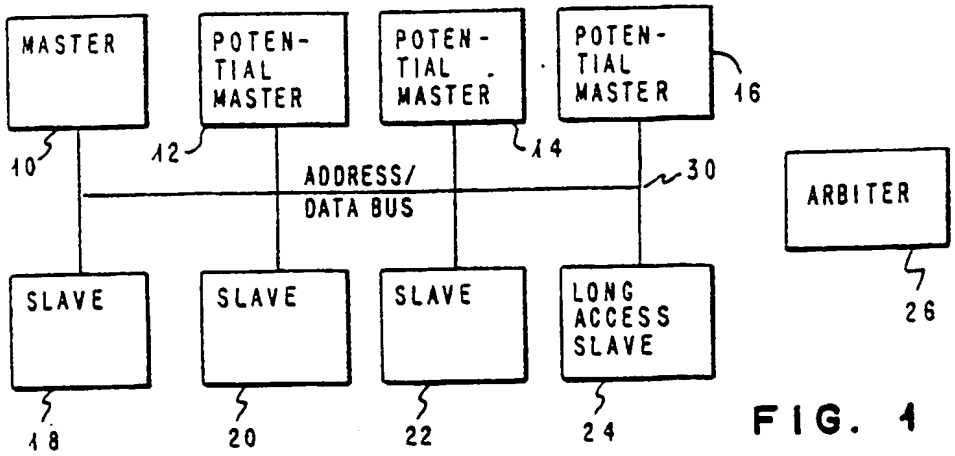


FIG. 1

EP 0 278 264 A3

European Patent Office

EUROPEAN SEARCH REPORT

Application Number

EP 88 10 0674

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int. Cl.5)
A	EP-A-0 138 676 (DIGITAL EQUIPMENT CORP.) * Abstract; page 4, line 24 - page 11, line 17 *	1-10	G 06 F 13/36 G 06 F 13/42
A	EP-A-0 094 140 (DATA GENERAL CORP.) * Abstract; page 2, line 11 - page 5, line 7; figure 1 *	1-10	
A	EP-A-0 114 485 (TEXAS INSTRUMENTS INC.) * Abstract; page 2, line 10 - page 3, line 26; page 7, line 9 - page 10, line 27; figures 1,3 *	1-10	

TECHNICAL FIELDS SEARCHED (Int. Cl.5)

G 06 F 13

The present search report has been drawn up for all claims

Place of search	Date of completion of search	Examiner
The Hague	19 October 90	MCDONAGH F.M.

CATEGORY OF CITED DOCUMENTS

X: particularly relevant if taken alone
Y: particularly relevant if combined with another document of the same catagory
A: technological background
O: non-written disclosure
P: intermediate document
T: theory or principle underlying the invention

E: earlier patent document, but published on, or after the filing date
D: document cited in the application
L: document cited for other reasons

&: member of the same patent family, corresponding document